

IDENTIFICATION

PRODUCT CODE:	MAINDEC-08-0G0PA-A-D
PRODUCT NAME:	DR8-EA DIAGNOSTIC (FOR USE ON SYSTEMS THAT HAVE A TRADITIONAL PDP-8 PROCESSOR WITH A DW8/E OMNIBUS CONVERTER
DATE REVISED:	JULY 3, 1973
MAINTAINER:	DIAGNOSTIC GROUP
AUTHOR:	MIKE HARE

COPYRIGHT (C) 1972, 1973, DIGITAL EQUIPMENT CORPORATION., MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED TO PURCHASER UNDER A LICENSE FOR USE  
ON A SINGLE SYSTEM AND CAN BE COPIED (WITH INCLUSION  
OF DEC'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT  
AS MAY OTHERWISE BE PROVIDED IN WRITING BY DEC.

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT  
NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL  
EQUIPMENT CORPORATION.

DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY  
OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DEC.



1. ABSTRACT

THIS DIAGNOSTIC HAS BEEN MODIFIED FOR TESTING THE DR8-EA INTERFACE ON SYSTEMS THAT HAVE A TRADITIONAL PDP-8 PROCESSOR WITH A DW8/E BUS CONVERTER.

THIS PROGRAM IS A DIAGNOSTIC AND EXERCISER FOR THE DR8-EA 12 CHANNEL BUFFERED DIGITAL INTERFACE. ALL FUNCTIONS ARE TESTED AND ERRORS ARE REPORTED BY HALTS AND/OR ERROR TYPEOUTS.

2. REQUIREMENTS

2.1 EQUIPMENT

TRADITIONAL PDP-8 COMPUTER WITH 4K OF CORE  
ASR-33 TELETYPE (OR EQUIVALENT)  
DW8/E BUS CONVERTER  
DR8-EA WITH TEST CABLE

2.2 STORAGE

THE PROGRAM USES LOCATIONS 0000-4121

3. LOADING PROCEDURE

THE STANDARD PROCEDURE FOR LOADING BINARY TAPES SHOULD BE USED.

4. STARTING PROCEDURE

4.1 STARTING ADDRESS

200-INPUT DEVICE CONFIGURATION  
201-START WITH STANDARD CONFIGURATION

4.2 SWITCH SETTINGS

FOR EITHER STARTING ADDRESS, NORMAL SETTING IS SR0-SR11-0 (DOWN).

4.3 PROGRAM AND/OR OPERATOR ACTION

LOAD PROGRAM INTO MEMORY  
SET SWITCH REGISTER TO DESIRED STARTING ADDRESS  
LOAD ADDRESS  
CLEAR SWITCHES  
PRESS CLEAR AND CONTINUE



#### 4.3.1 FOR STARTING ADDRESS 200

THE PROGRAM WILL TYPE "SET SR FOR DEVICE CODE AND CONT"  
AND THEN HALT.

SET SWITCHES TO 000X WHERE X IS AN OCTAL  
NUMBER CORRESPONDING TO THE 3 LSB OF THE DEVICE SELECTOR CODE,  
E.G. IF DEVICE CODE IS 53, SET SR TO 0003,  
PRESS CONTINUE.

PROGRAM WILL RESPOND BY TYPING

"SET SR FOR INTERRUPT JUMPERS AND CONT" AND THEN HALT.  
SET SWITCHES FOR ALL INPUT REGISTER BITS JUMPED TO INTERRUPT.  
PRESS CONTINUE.

PROGRAM WILL RESPOND BY TYPING

"SET SR FOR FLIPFLOP JUMPERS AND CONT" AND THEN HALT.  
SET SWITCHES FOR ALL INPUT REGISTER FLIPFLOPS.  
PRESS CONTINUE.

PROGRAM WILL RESPOND BY TYPING

"SET SR FOR RUN" AND THEN HALT.  
SET SWITCHES AS DESIRED. (REFER TO SECTION 5.1)  
PRESS CONTINUE.

PROGRAM WILL BEGIN TEST EXECUTION

#### 4.3.2 FOR STARTING ADDRESS 201

SET SWITCHES AS DESIRED. (REFER TO SECTION 5.1)  
DEPRESS CONTINUE.

PROGRAM WILL BEGIN TEST EXECUTION



5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

SR0=1, SUPPRESS ERROR HALT  
SR1=1, SUPPRESS ERROR TYPEOUT  
SR2=1, LOOP ON CURRENT TEST  
SR3=1, LOOP WITH CURRENT DATA  
SR4=1, SUPPRESS BELL OR TYPEOUT AT END OF PASS  
SR5=1, SUPPRESS ITERATIONS  
SR6=1, ESCAPE TO NEXT TEST ON ERROR

5.2 PROGRAM AND/OR OPERATOR ACTION

5.2.1.1 WITH SWITCHES SET AS IN 4.2, THE PROGRAM WILL RUN ALL TESTS SEQUENTIALLY, EACH 10T TEST WILL BE REPEATED 4096 TIMES. EACH DATA TEST WILL BE REPEATED 50 TIMES. AFTER ALL TESTS HAVE BEEN COMPLETED, THE PROGRAM WILL TYPE "DR" AND START ALL TESTS AGAIN.  
IF AN ERROR OCCURS, THE PROGRAM WILL HALT AND TYPE AN APPROPRIATE ERROR MESSAGE (SEE SECTION 6 FOR DETAILS).

5.2.1.2 WITH SR0=1 (UP), PROGRAM ACTION WILL BE AS IN 5.2.1.1, EXCEPT NO TYPEOUT WILL OCCUR.

5.2.1.3 WITH SR2=1(UP) PROGRAM ACTION WILL BE AS IN 5.2.1.1, EXCEPT NO TYPEOUT WILL OCCUR. THE ADDRESS OF THE FAILING TEST WILL BE DISPLAYED IN THE COMPUTER AC.

5.2.1.4 WITH SR4=1(UP), PROGRAM ACTION WILL BE AS IN 5.2.1.1 EXCEPT NO END OF PASS TYPEOUT WILL OCCUR.

5.2.1.5 WITH SR5=1 (UP), EACH TEST WILL BE EXECUTED ONLY ONCE. INSTEAD OF TYPING "DR", THE PROGRAM WILL RING THE TTY BELL

5.2.1.6 WITH SR0=1 AND SR6=1, PROGRAM ACTION WILL BE AS IN 5.2.1.1 IF NO ERRORS OCCUR.  
IF AN ERROR OCCURS, THE PROGRAM WILL TYPE AN APPROPRIATE ERROR MESSAGE AND IMMEDIATELY TERMINATE ITERATIONS OF THE FAILING TEST. THE PROGRAM WILL THEN START THE NEXT TEST IN SEQUENCE.

6. ERRORS

6.1 NORMAL OPERATION

IF AN ERROR OCCURS WITH SWITCHES SET AS IN 4.2, THE PROGRAM WILL TYPE AN APPROPRIATE ERROR MESSAGE (WITH DATA IF APPLICABLE) AND HALT.

THE FORMAT OF THE ERROR TYPEOUT IS

XXXX MESSAGE  
HEADER FOR DATA (IF APPLICABLE)  
DATA (IF APPLICABLE)

WHERE XXXX= ADDRESS OF JMS TO ERROR ROUTINE IN TEST THAT FAILED.

6.2 ERROR RECOVERY

SET SR0=1(UP) TO ESCAPE TO NEXT TEST, PRESS CONTINUE.

6.3 ERROR LOOP (LOTS)

SET SR0=1 TO SUPPRESS HALT  
SET SR1=1 TO SUPPRESS TYPEOUT  
SET SR2=1 TO LOOP ON CURRENT FAILING TEST

6.4 ERROR LOOP (DATA)

SAME AS 6.3 EXCEPT USE SR3 INSTEAD OF SR2 TO LOOP WITH CURRENT DATA.

7. RESTRICTIONS

7.1 STARTING RESTRICTIONS

TEST JUMPER CABLE MUST BE INSTALLED.  
ANY FLOATING INPUTS TO INPUT REGISTER SHOULD BE GROUNDED, OR ERRORS MAY OCCUR.

7.2 OPERATING RESTRICTIONS

NONE



8. MISCELLANEOUS

8.1 EXECUTION TIME

EXECUTION TIME IS APPROXIMATELY 9 MINUTES FOR FULL ITERATION AND APPROXIMATELY 10 SECONDS WITH ITERATIONS SUPPRESSED.

9. PROGRAM DESCRIPTION

THE DR8-EA IS A TEST OF ALL FUNCTIONS OF THE INTERFACE.

THE PROGRAM SEQUENCE IS AS FOLLOWS:

ALL BASIC IOT TESTS ARE EXECUTED 4096 TIMES.  
ALL OUTPUT REGISTER FUNCTIONS ARE TESTED WITH BINARY COUNT PATTERNS.  
ALL INPUT REGISTER FUNCTIONS ARE TESTED USING BINARY COUNT PATTERNS.  
INTERACTION BETWEEN INPUT AND OUTPUT REGISTERS IS TESTED FOR  
WITH BINARY COUNT PATTERNS.  
ALL SKIPS AND INTERRUPTS ARE TESTED USING BINARY COUNT PATTERNS.

10. LISTING



/MAINDEC-08-DCORA-A DR8-EA DIAGNOSTIC FOR TRADITIONAL PDP8 PROCESSORS  
 /WITH A DW8/E OMNIBUS CONVERTER  
 /COPYRIGHT 1972,1973 DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

/STARTING ADDRESS: 200-INPUT DEVICE CONFIGURATION  
 / 201-USE STANDARD CONFIGURATION

/SWITCH REGISTER OPTIONS

/SR00 =1, SUPPRESS HALT ON ERROR  
 /SR01 =1, SUPPRESS ERROR TYPEOUT  
 /SR02 =1, LOOP ON CURRENT TEST  
 /SR03 =1, LOOP WITH CURRENT DATA  
 /SR04 =1, SUPPRESS BELL AT END OF PASS  
 /SR05 =1, SUPPRESS ITERATIONS  
 /SR06 =1, ESCAPE TO NEXT TEST ON ERROR

/INSTRUCTION DEFINITIONS

3114 MQL=DCA MQ  
 4115 MQA=JMS SIMMGA  
 4046 BSW=JMS SIMBSW  
 4063 CAF=JMS SIMCAF  
 4076 SRQ=JMS SIMSRQ  
 4432 DBEI=JMS I XDBEI  
 4433 ODSK=JMS I XODSK  
 4434 DBCI=JMS I XDBCI  
 4435 DBRI=JMS I XDBRI  
 4436 DBCO=JMS I XDBCO  
 4437 ODSO=JMS I XODSO  
 4440 DBRO=JMS I XDBRO

/LOCATION EQUIVALENCIES

/ MSTDGT=ERADR+1  
 / LSTDGT=ERADR+2

/GENERAL VARIABLES

JMP I 2

0001 \*1  
 5402  
 0010 \*20  
 0000 POINT1, 0  
 0020 \*20  
 3002 CNT1,  
 0021 DATA1, 0  
 0022 DATA2, 0  
 0023 DATA3, 0  
 0024 DATA4, 0  
 0025 DATA5, 0  
 0026 IJUMPE, 7777  
 7777  
 0027 FJUMPE, 7777  
 0030 TYPFLG, 0

/INTERRUPT JUMPER MASK

/FLIPFLOP JUMPER MASK

0031	0000	LPCNT, 0	/INDIRECT POINTERS
0032	0266	XDBE1,	DBEIX
0033	0273	XDBSK,	DBSKX
0034	0300	XDBCI,	DBCI X
0035	0305	XDBRI,	DBRI X
0036	0312	XDBCO,	DBCOX
0037	0317	XDBSO,	DBSOX
0040	0324	XDBRO,	DBROX
0041	3200	XPRINT,	PRINT
0042	3251	XTYPE,	TYPE
0043	2600	XERROR,	ERROR
0044	2667	XLOOP1,	LOOP1
0045	2712	XLOOP2,	LOOP2
0046	0000	SIMBSW,	0
0047	3113	DCA	TYE1
0050	7012	RTR	
0051	7012	RTR	
0052	7012	RTR	
0053	1113	TAD	TYE1
0054	0062	AND	K7700
0055	1113	TAD	TYE1
0056	7006	RTL	
0057	7006	RTL	
0060	7006	RTL	
0061	5446	JMP I	SIMBSW
0062	7700	K7700,	7700
0063	0000	SIMCAP,	0
0064	0002	IOF	CMA
0065	7240	CLA	CMA
0066	4434	DBCI	
0067	7410	SKP	
0070	7402	HLT	
0071	4436	DBCO	
0072	7410	SKP	
0073	7402	HLT	
0074	7300	CLA	CLL
0075	5463	JMP I	SIMCAF
0076	3000	SIMSRQ,	0
0077	3113	DCA	TYE1
2100	1177	TAC	(RETURN
0101	3002	DCA	2
0102	6001	ION	
0103	7000	NOP	
0104	6002	IOF	
0105	7410	SKP	
0106	2076	RETURN,	ISZ
0107	7324	CLA	CLL
0110	3002	DCA	2
0111	1113	TAD	TYE1
			/NO INTERRUPT

/DBCI SKIPPED

/DBCO SKIPPED







0246 0400 INIT1

/INITIALIZATION CONSTANTS AND VARIABLES  
/

/BASIC IOT

0247 0000 IOTS,  
0250 0262 TIOT,  
0251 0267 DBDIX+1  
0252 0274 DBEIX+1  
0253 0301 DBSKX+1  
2254 0306 DBCIX+1  
0255 0313 DBRIX+1  
0256 0320 DBCOX+1  
0257 0325 DBSOX+1  
0260 3650 DBROX+1  
DCA I TIOT

/IOT SUBROUTINES  
/

/DISABLE DATA BUFFER INTERRUPT (DBDI,65X0)

0261 0000 DBDIX,  
0262 6500 6500

SKP /TRAP FOR UNDESIRE  
HLT /SKIPS  
JMP I DBDIX

/ENABLE DATA BUFFER INTERRUPTS (DBEI,65X1)

0266 0000 DBEIX,  
0267 6501 6501

SKP /TRAP FOR UNDESIRE  
HLT /SKIPS  
JMP I DBEIX

/SKIP ON DATA BUFFER INPUT FLAG (DBSK,65X2)

0273 0000 DBSKX,  
0274 6502 6502

0275 7410 SKP DBSKX  
0276 2273 IZ DBSKX  
0277 5673 JMP I DBSKX

/OS TO INPUT REGISTER CORRESPONDING  
/TO IS IN AC (DBCI,65X3)

0300 0000 DBCIX,  
0301 6503 6503

SKP /TRAP FOR UNDESIRE  
HLT /SKIPS  
JMP I DBCIX

```

0305 0000
0306 6504
    DBRIX,
/INPUT REGISTER TO AC (DBRI,65X4)
/
0307 7410
0310 7402
0311 5705
    SKP
    HLT
    JMP I DBRIX
/TRAP FOR UNDESIRE
/
0312 0000
0313 6505
    DBCOX,
/ZEROS TO OUTPUT REGISTER CORRESPONDING TO
/ONES IN AC (DBCO,65X5)
/
0314 7410
0315 7402
0316 5712
    SKP
    HLT
    JMP I DBCOX
/TRAP FOR UNDESIRE
/
0317 0000
0320 6506
    DBSOX,
/IS TO OUTPUT REGISTER CORRESPONDING
/TO 1S IN AC (DBSO,65X6)
/
0321 7410
0322 7402
0323 5717
    SKP
    HLT
    JMP I DBSOX
/TRAP FOR UNDESIRE
/
0324 0000
0325 6507
    DBROX,
/JAM TRANSFER OUTPUT REGISTER TO AC (DBRO 65X7)
/
0326 7410
0327 7402
0330 5724
0375 7770
0376 6500
0377 0007
    SKP
    HLT
    JMP I DBROX
/TRAP FOR UNDESIRE
/
    PAGE
0400 3030
0401 3031
0402 4063
0403 4440
0404 3021
0405 1021
0406 7650
/IS OUTPUT REGISTER CLEARED BY INITIALIZE?
/
INIT1, DCA
    TYPFLG
    LPCNT
/CLEAR ERROR FLAG
/SET ITERATION COUNT TO 4096(DECIMAL)
/INITIALIZE INTERFACE
/READ OUTPUT REGISTER
/SAVE REGISTER DATA
/GET REGISTER DATA
/WAS REGISTER CLEARED BY INITIALIZE

```



```

0407 5214 JMP .+5
0410 4443 JMS I XERROR
0411 3645 INIT1E=1
0412 3515 DH1=1
0413 7777 -1
0414 4444 JMS I XLOOP1
0415 5202 JMP INIT1+2
/
/IS INPUT REGISTER CLEARED BY INITIALIZE?
/
INIT2,
0416 3030 DCA TYPFLG
0417 3031 DCA LPCNT
0420 7410 SKP
0421 4063 CAF
0422 4435 DBR1
0423 3021 DCA DATA1
0424 1021 TAD DATA1
0425 7650 SNA CLA
0426 5233 JMP .+5
0427 4443 JMS I XERROR
0430 3661 INIT2E=1
0431 3515 DH1=1
0432 7777 -1
0433 4444 JMS I XLOOP1
0434 5221 JMP INIT2+3
/
/IS SKIP FLAG SET AFTER INITIALIZE
/
INIT3,
0435 3030 DCA TYPFLG
0436 3031 DCA LPCNT
0437 7410 SKP
0440 4063 CAF
0441 4432 DBE1
0442 4433 DBSK
0443 5250 JMP .+5
0444 4443 JMS I XERROR
0445 3674 INIT3E=1
0446 3514 DH0=1
0447 0000 0
0450 4444 JMS I XLOOP1
0451 5240 JMP INIT3+3
/
/DOES OUTPUT REGISTER JAM TRANSFER TO AC?
/
TRAN1,
0452 3030 DCA TYPFLG
0453 3031 DCA LPCNT
0454 4063 CAF
0455 7340 CLA CLL CMA
0456 4440 DBR0
0457 3021 DCA DATA1
0460 1021 TAD DATA1
0461 7650 SNA CLA
0462 5267 JMP .+5
0463 4443 JMS I XERROR

```



```

0464 3703      TRAN1E=1      /DBRO DID NOT CLEAR AC"
0465 3525      DH2=1        /"AC CONTENTS"
0466 7777      -1          /NUMBER OF WORDS TO BE OUTPUT
0467 4444      JMS I        /CHECK FOR LOOP ON CURRENT TEST
0470 5254      JMP         /LOOP ON CURRENT TEST
/
/DOES INPUT REGISTER JAM TRANSFER TO AC
/
TRAN2,
0471 3030      DCA         /CLEAR ERROR FLAG
0472 3031      DCA         /SET ITERATION COUNT TO 4096(DECIMAL)
0473 4063      CAF         /INITIALIZE INTERFACE
0474 7340      CLA CLL CMA /SET AC =7777
0475 4435      DBRI       /READ INPUT REGISTER
0476 3021      DCA         /SAVE AC CONTENTS
0477 1021      TAD DATA1 /GET AC CONTENTS
0500 7650      SNA CLA    /WAS AC CLEARED BY TRANSFER
0501 5306      JMP .+5     /DATA CORRECT, CONTINUE
0502 4443      JMS I      /NO, ERROR
0503 3716      TRAN2E=1   /DBRI DID NOT CLEAR AC"
0504 3525      DH2=1      /"AC CONTENTS"
0505 7777      -1        /NUMBER OF WORDS TO BE OUTPUT
0506 4444      JMS I      /CHECK FOR LOOP ON CURRENT TEST
0507 5273      JMP         /LOOP ON CURRENT TEST
/
/DOES TRANSFER TO OUTPUT REGISTER CHANGE AC (WITH AC=7777,DBSO)
/
TRAN3,
0510 3030      DCA         /CLEAR ERROR FLAG
0511 3031      DCA         /SET ITERATION COUNT TO 4096(DECIMAL)
0512 4063      CAF         /INITIALIZE INTERFACE
0513 7340      CLA CLL CMA /SET AC =7777
0514 4437      DBSO       /BIT SET OUTPUT REGISTER
0515 3021      DCA         /SAVE AC CONTENTS
0516 1021      TAD DATA1 /GET AC CONTENTS
0517 7040      CMA CLA    /COMPLIMENT DATA TO TEST FOR 7777
0520 7650      SNA CLA    /DID AC CHANGE
0521 5326      JMP .+5     /DATA CORRECT, CONTINUE
0522 4443      JMS I      /NO, ERROR
0523 3731      TRAN3E=1   /DBSO CHANGED AC"
0524 3525      DH2=1      /"AC CONTENTS"
0525 7777      -1        /NUMBER OF WORDS TO BE OUTPUT
0526 4444      JMS I      /CHECK FOR LOOP ON CURRENT TEST
0527 5312      JMP         /LOOP ON CURRENT TEST
/
/DOES TRANSFER TO OUTPUT REGISTER CHANGE AC (WITH AC=7777, DBCO)
/
TRAN4,
0530 3030      DCA         /CLEAR ERROR FLAG
0531 3031      DCA         /SET ITERATION COUNT TO 4096(DECIMAL)
0532 4063      CAF         /INITIALIZE INTERFACE
0533 7340      CLA CLL CMA /SET AC =7777
0534 4436      DBCO       /BIT CLEAR OUTPUT REGISTER
0535 3021      DCA         /SAVE AC CONTENTS
0536 1021      TAD DATA1 /GET AC CONTENTS
0537 7040      CMA CLA    /COMPLIMENT DATA TO TEST FOR 7777
0540 7650      SNA CLA    /DID AC CHANGE

```



```

0541 5346 JMP .+5
0542 4443 JMS I XERROR
0543 3741 TRANSF=1
0544 3525 DH2=1
0545 7777 -1
0546 4444 JMS I XLOOP1
0547 5332 JMP TRANSF+2
/DOES TRANSFER TO INPUT REGISTER CHANGE AC (WITH AC=7777, DBCI)
/
TRAN5,
0550 3030 DCA TYPFLG
0551 3031 DCA LPCNT
0552 4063 CAF
0553 7340 CLA CLL CHA
0554 4434 DBCI
0555 3021 DCA DATA1
0556 1021 TAD DATA1
0557 7040 CMA CLA
0558 7650 SNA CLA
0559 5366 JMP .+5
0560 4443 JMS I XERROR
0561 3751 TRANSF=1
0562 3525 DH2=1
0563 7777 -1
0564 4444 JMS I XLOOP1
0565 5352 JMP TRANS+2
0566 5777 JMP TRANS6
0577 0600
PAGE

```

```

/DOES TRANSFER TO OUTPUT REGISTER CHANGE AC (WITH AC=0,DBSO)
/
TRAN6,
0600 3030 DCA TYPFLG
0601 3031 DCA LPCNT
0602 4063 CAF
0603 4437 DBSO
0604 3021 DCA DATA1
0605 1021 TAD DATA1
0606 7650 SNA CLA
0607 5214 JMP .+5
0608 4443 JMS I XERROR
0609 3731 TRANSF=1
0610 3525 DH2=1
0611 7777 -1
0612 4444 JMS I XLOOP1
0613 5202 JMP TRANS6+2
/DOES TRANSFER TO OUTPUT REGISTER CHANGE AC (WITH AC=0,DBCO)
/
TRAN7,
0616 3030 DCA TYPFLG
0617 3031 DCA LPCNT
0618 4063 CAF
0619 4436 DBCO

```



0622	3021	DCA	DATA1	/SAVE AC CONTENTS
0623	1021	TAD	DATA1	/GET AC CONTENTS
0624	7650	SNA CLA		/IS AC STILL 0
0625	5232	JMP	.+5	/DATA CORRECT, CONTINYE
0626	4443	JMS I	XERROR	/NO, ERROR
0627	3741	TRAN4E-1		/WDBCO CHANGED AC"
0630	3525	DH2-1		/AC CONTENTS"
0631	7777	-1		/NUMBER OF WORDS TO BE OUTPUT
0632	4444	JMS I	XLOOP1	/CHECK FOR LOOP ON CURRENT TEST
0633	5220	JMP	TRAN7+2	/LOOP ON CURRENT TEST
		/		/DOES TRANSFER TO INPUT REGISTER CHANGE AC (WITH AC=0, DBCI)
		/		
0634	3030	TRAN8,	TYPEFLG	/CLEAR ERROR FLAG
0635	3031	DCA	LPCNT	/SET ITERATION COUNT TO 4096 (DECIMAL)
0636	4063	CAF		/INITIALIZE INTERFACE
0637	4434	DBCI		/BIT CLEAR INPUT REGISTER
0640	3021	DCA	DATA1	/SAVE AC CONTENTS
0641	1021	TAD	DATA1	/GET AC CONTENTS
0642	7650	SNA CLA		/IS AC STILL 0
0643	5250	JMP	.+5	/DATA CORRECT, CONTINUE
0644	4443	JMS I	XERROR	/AC CHANGED, ERROR
0645	3751	TRAN5E-1		/WDBCI CHANGED AC"
0646	3525	DH2-1		/AC CONTENTS"
0647	7777	-1		/NUMBER OF WORDS TO BE OUTPUTED
0650	4444	JMS I	XLOOP1	/CHECK FOR LOOP ON CURRENT TEST
0651	5236	JMP	TRAN8+2	/LOOP ON CURRENT TEST
0652	5777	JMP	OUT1	/GO TO NEXT TEST
0777	1000			

PAGE

1000	3030	OUT1,	TYPEFLG	/CLEAR ERROR FLAG
1001	3031	DCA	LPCNT	/SET ITERATION COUNT TO 4096(DECIMAL)
1002	4063	CAF		/INITIALIZE INTERFACE
1003	7340	CLA CLL CMA		/SET AC 7777
1004	4437	DBSO		/BIT SET OUTPUT REGISTER
1005	4440	DBRO		/READ OUTPUT REGISTER
1006	3021	DCA	DATA1	/SAVE REGISTER DATA
1007	1021	TAD	DATA1	/GET REGISTER DATA
1010	7010	CMA CLA		/COMPLEMENT DATA TO TEST FOR 7777
1011	7650	SNA CLA		/IS REGISTER=7777
1012	5227	JMP	.+5	/DATA CORRECT, CONTINUE
1013	4443	JMS I	XERROR	/NO, ERROR
1014	3761	OUT1E-1		/WDBSO ERROR"
1015	3515	DH1-1		/REGISTER DATA"
1016	7777	-1		/NUMBER OF WORDS TO BE OUTPUT
1017	4063	CAF		/INITIALIZE INTERFACE
1020	4440	DBRO		/READ OUTPUT REGISTER
1021	3021	DCA	DATA1	/SAVE REGISTER DATA
1022	1021	TAD	DATA1	/GET REGISTER DATA
1023	7650	SNA CLA		/WAS REGISTER CLEARED



```

1024 5231 JMP .+5
1025 4443 JMS I XERROR
1026 3645 INITIE=1
1027 3515 DH1=1
1030 7777 -1
1031 4444 JMS I XLOOP1
1032 5202 JMP OUT1+2
/
/ CAN ALL BITS OF OUTPUT REGISTER BE CLEARED (DBCO)
/
OUT2,
1033 3030 OCA TYPFLG
1034 3031 OCA LPCNT
1035 7340 CLA CLL CMA
1036 4437 DBSO
1037 4436 DBCO
1040 7300 CLA CLL
1041 4440 DBRO
1042 3021 OCA DATA1
1043 1021 TAD DATA1
1044 7630 SNA CLA
1045 5252 JMP .+5
1046 4443 JMS I XERROR
1047 3645 INITIE=1
1050 3515 DH1=1
1051 7777 -1
1052 4444 JMS I XLOOP1
1053 5235 JMP OUT2+2
/
/ CAN EACH BIT OF OUTPUT REGISTER BE SET
/ INDEPENDENTLY (DBSO)
/
OUT3,
1054 3030 OCA TYPFLG
1055 1176 TAD C=62
1056 3031 OCA LPCNT
1057 3021 OCA DATA1
OUT3A,
1060 4063 CAF
1061 1021 TAD DATA1
1062 4437 DBSO
1063 7300 CLA CLL
1064 4440 DBRO
1065 3022 OCA DATA2
1066 1021 TAD DATA1
1067 7041 CIA
1070 1022 TAD DATA2
1071 7650 SNA CLA
1072 5277 JMP .+5
1073 4443 JMS I XERROR
1074 3761 OUTIE=1
1075 3560 DH4=1
1076 7776 -2
1077 4445 JMS I XLOOP2
1100 5260 JMP OUT3A
1101 2021 ISZ DATA1
1102 5260 JMP OUT3A
/ DATA CORRECT, CONTINUE
/ NO, ERROR
/ "OUTPUT REG NOT CLEARED"
/ "REGISTER DATA"
/ NUMBER OF WORDS TO BE OUTPUT
/ CHECK FOR LOOP ON CURRENT TEST
/ LOOP ON CURRENT TEST
/ CLEAR ERROR FLAG
/ SET ITERATION COUNT TO 4096(DECIMAL)
/ SET AC =7777
/ BIT SET OUTPUT REGISTER
/ BIT CLEAR OUTPUT REGISTER
/ READ OUTPUT REGISTER
/ SAVE REGISTER DATA
/ GET REGISTER DATA
/ WAS OUTPUT REGISTER CLEARED
/ DATA CORRECT, CONTINUE
/ NO, ERROR
/ "OUTPUT REGISTER NOT CLEARED"
/ "REGISTER DATA"
/ NUMBER OF WORDS TO BE OUTPUT
/ CHECK FOR LOOP ON CURRENT TEST
/ LOOP ON CURRENT TEST
/ CLEAR ERROR FLAG
/ SET ITERATION COUNT
/ TO 50(DECIMAL)
/ CLEAR TEST DATA
/ INITIALIZE INTERFACE
/ GET TEST DATA
/ BIT SET OUTPUT REGISTER
/ READ OUTPUT REGISTER
/ SAVE REGISTER DATA
/ GET TEST DATA
/ COMPARE TO REGISTER CONTENTS
/ DO THEY COMPARE
/ DATA CORRECT, CONTINUE
/ NO, ERROR
/ "DBSO ERROR"
/ "EXPECTED RECEIVED"
/ NUMBER OF WORDS TO BE OUTPUT
/ TEST FOR LOOP ON SAME DATA, ESCAPE ON DATA ERROR
/ LOOP WITH SAME DATA
/ INCREMENT DATA PATTERN
/ CONTINUE TEST

```



1103	4444	JMS I	XLOOP1	/CHECK FOR LOOP ON CURRENT TEST
1104	5257	JMP	OUT3+3	/LOOP ON CURRENT TEST
				/CAN EACH BIT OF OUTPUT REGISTER BE CLEARED
				/INDEPENDENTLY (DBCO)
				/
1105	3030	OUT4,	DCA	/CLEAR ERROR FLAG
1106	1176		TAD	/SET ITERATION COUNT
1107	3031		C=62	/TO 50(DECIMAL)
1110	3021		DCA	/CLEAR TEST DATA
1111	4063	OUT4A,	CAF	/INITIALIZE INTERFACE
1112	1021		TAD	/GET MASK
1113	7040		CMA	/COMPLEMENT TO GET EXPECTED RESULT
1114	3022		DCA	/SAVE EXPECTED RESULT
1115	7040		CMA	/SET OUTPUT REGISTER TO 7777
1116	4437		DMSO	/BIT SET OUTPUT REGISTER
1117	7300		CLA GLL	
1120	1021		TAD	/GET PATTERN TO CLEAR OUTPUT REGISTER
1121	4436		DBCO	/BIT CLEAR OUTPUT REGISTER
1122	4440		DBRO	/READ OUTPUT REGISTER
1123	3023		DCA	/SAVE REGISTER DATA
1124	1022		TAD	/GET EXPECTED RESULT
1125	7041		CIA	
1126	1023		TAD	/COMPARE TO RECEIVED DATA
1127	7650		SNA CLA	/WHERE CORRECT BITS IN OUTPUT REGISTER CLEARED
1130	5335		JMS I	/DATA CORRECT, CONTINUE
1131	4443		JMP	/NO, ERROR
1132	3767		OUT4E-1	/DBCO ERROR
1133	3934		DH3-1	/MASK EXPECTED RECEIVED
1134	7775		-3	/NUMBER OF WORDS TO BE OUTPUT
1135	4445		JMS I	/TEST FOR LOOP ON SAME DATA, ESCAPE ON DATA ERROR
1136	5311		JMP	/LOOP WITH SAME DATA
1137	2021		ISE	/INCREMENT DATA PATTERN
1140	5311		JMP	/CONTINUE TEST
1141	4444		JMS I	/CHECK FOR LOOP ON CURRENT TEST
1142	5310		JMP	/LOOP ON CURRENT TEST
1143	5777		JMP	
1177	1200			

PAGE

				/WITH OUTPUT REGISTER CLEARED, DOES CLEARING
				/OUTPUT REGISTER CHANGE ANY BIT IN OUTPUT
				/
1200	3030	OUT5,	DCA	/CLEAR ERROR FLAG
1201	1176		TAD	/SET ITERATION COUNT
1202	3031		DCA	/TO 50 (DECIMAL)
1203	3021		DCA	/CLEAR TEST DATA
1204	3022		DCA	/CLEAR EXPECTED RESULT
1205	4063	OUT5A,	CAF	/INITIALIZE INTERFACE
1206	1021		TAD	/GET TEST DATA
1207	4436		DBCO	/BIT CLEAR OUTPUT REGISTER
1210	7300		CLA GLL	
1211	4440		DBRO	/READ OUTPUT REGISTER
1212	3023		DCA	/SAVE REGISTER DATA
1213	1023		TAD	/GET REGISTER DATA



```

1214 7650 SNA CLA
1215 5222 JMP .+5
1216 4443 JMS I XERROR
1217 3767 OUT4E-1
1220 3534 DH3-1
1221 7775 -3
1222 4445 JMS I XLOOP2
1223 5205 JMP OUT5A
1224 2021 ISZ DATA1
1225 5205 JMP OUT5A
1226 4444 JMS I XLOOP1
1227 5203 JMP OUT5+3

```

```

/IS OUTPUT REGISTER 0
/DOES SETTING OUTPUT REGISTER TWICE WITH SAME
/NO, ERROR
/DBCO ERROR"
/MASK EXPECTED RECEIVED"
/NUMBER OF DATA WORDS
/TEST FOR LOOP ON SAME DATA
/LOOP WITH SAME DATA
/INCREMENT DATA PATTERN
/CONTINUE
/CHECK FOR LOOP ON CURRENT TEST
/LOOP ON CURRENT TEST

```

```

/DOES SETTING OUTPUT REGISTER TWICE WITH SAME
/DOES READING OUTPUT REGISTER TWICE CHANGE
/OUTPUT REGISTER

```

```

1230 3030 OUT6,
1231 1176 DCA TYPFLG
1232 3031 TAD C-62
1233 3021 DCA LPCNT
1234 4063 DCA DATA1
1235 1021 CAF DATA1
1236 4437 TAD DATA1
1237 4437 DBSO
1240 7300 DBSO
1241 4440 CLA CLL
1242 3022 DBRO
1243 1021 DCA DATA2
1244 7041 TAD DATA1
1245 1022 CIA DATA2
1246 7650 SNA CLA
1247 5254 JMP .+5
1250 4443 JMS I XERROR
1251 3767 OUT4E-1
1252 3560 DH4-1
1253 7775 -2
1254 4445 JMS I XLOOP2
1255 5234 JMP OUT6A
1256 2021 ISZ DATA1
1257 5234 JMP OUT6A
1260 4444 JMS I XLOOP1
1261 5233 JMP OUT6+3

```

```

/CLEAR ERROR FLAG
/SET ITERATION COUNT
/TO 50 (DECIMAL)
/CLEAR TEST DATA
/INITIALIZE INTERFACE
/GET TEST DATA
/BIT SET OUTPUT REGISTER
/BIT SET OUTPUT REGISTER
/READ OUTPUT REGISTER
/SAVE REGISTER DATA
/GET TEST DATA
/COMPARE TO REGISTER DATA
/ARE THEY THE SAME
/DOES SETTING OUTPUT REGISTER TWICE WITH SAME
/NO, ERROR
/DBSO ERROR"
/MASK EXPECTED RECEIVED"
/NUMBER OF DATA WORDS
/TEST FOR LOOP ON SAME DATA
/LOOP WITH SAME DATA
/INCREMENT DATA PATTERN
/CONTINUE
/CHECK FOR LOOP ON CURRENT TEST
/LOOP ON CURRENT TEST

```

```

/DOES SETTING OUTPUT REGISTER TWICE WITH SAME
/DOES READING OUTPUT REGISTER TWICE CHANGE
/OUTPUT REGISTER

```

```

1262 3030 OUT7,
1263 1176 DCA TYPFLG
1264 3031 TAD C-62
1265 3021 DCA LPCNT
1266 4063 DCA DATA1
1267 1021 CAF DATA1
1270 4437 TAD DATA1

```

```

/CLEAR ERROR FLAG
/SET ITERATION COUNT
/TO 50 (DECIMAL)
/CLEAR TEST DATA
/INITIALIZE INTERFACE
/GET TEST DATA
/BIT SET OUTPUT REGISTER

```



1271	7300	CLA CLL	
1272	4440	DBRO	/READ OUTPUT REGISTER
1273	7300	CLA CLL	
1274	4440	DBRO	/READ OUTPUT REGISTER
1275	3022	DCA	/SAVE REGISTER DATA
1276	1021	TAD	/GET TEST DATA
1277	7041	CIA	
1300	1022	TAD	
1301	7650	SNA CLA	/COMPARE TO REGISTER DATA
1302	5307	JMP	/ARE THEY THE SAME
1303	4443	JMS I	/DATA CORRECT, CONTINUE
1304	3775	OUT7E-1	/NO, ERROR
1305	3560	DH4-1	/"DBRO ERROR"
1306	7776	=2	/"EXPECTED RECEIVED"
1307	4445	JMS I	/NUMBER OF DATA WORDS
1310	5266	JMP	/TEST FOR LOOP WITH SAME DATA
1311	2021	ISZ	/LOOP WITH SAME DATA
1312	5266	JMP	/INCREMENT DATA PATTERN
1313	4444	JMS I	/CONTINUE
1314	5265	JMP	/CHECK FOR LOOP ON CURRENT TEST
			/LOOP ON CURRENT TEST

1315	3030	DCA	/CLEAR ERROR FLAG
1316	1176	TAD	/SET ITERATION COUNT
1317	3031	DCA	/TO 50 (DECIMAL)
1320	3021	DCA	/CLEAR TEST DATA
1321	3022	DCA	/CLEAR EXPECTED RESULT
1322	4063	CAF	/INITIALIZE INTERFACE
1323	1021	TAD	/GET TEST DATA
1324	4437	DBSO	/BIT SET OUTPUT REGISTER
1325	4436	DBCO	/BIT CLEAR OUTPUT REGISTER
1326	4436	DBCO	/BIT CLEAR OUTPUT REGISTER
1327	7300	CLA CLL	
1330	4440	DBRO	/READ OUTPUT REGISTER
1331	3023	DCA	/SAVE REGISTER DATA
1332	1023	TAD	/GET REGISTER DATA
1333	7650	SNA CLA	/IS REGISTER 0
1334	5341	JMP	/DATA CORRECT, CONTINUE
1335	4443	JMS I	/NO, ERROR
1336	3767	OUT4E-1	/"DBCO ERROR"
1337	3234	DH3-1	/"MASK EXPECTED RECEIVED"
1340	7775	=3	/NUMBER OF DATA WORDS
1341	6445	JMS I	/TEST FOR LOOP WITH SAME DATA
1342	5322	JMP	/LOOP WITH SAME DATA
1343	2021	ISZ	/INCREMENT DATA PATTERN
1344	5322	JMP	/CONTINUE
1345	4444	JMS I	/CHECK FOR LOOP ON CURRENT TEST
1346	5320	JMP	/LOOP ON CURRENT TEST
1347	5777	JMP	
1377	1400		



```

1400 3030      IN1,      /CAN ALL BITS IN INPUT REGISTER BE SET
1401 3031      DCA      /DOES INITIALIZE CLEAR INPUT REGISTER
1402 4063      DCA      /CLEAR ERROR FLAG
1403 7340      CAF      /SET ITERATION COUNT TO 4096(DECIMAL)
1404 4437      CLA CLL CMA /INITIALIZE INTERFACE
1405 7300      DBSO      /SET AC =7777
1406 4435      CLA CLL      /BIT SET OUTPUT REGISTER
1407 3021      DBRI      /READ INPUT REGISTER
1410 1021      DCA      DATA1 /SAVE REGISTER DATA
1411 7040      TAD      DATA1 /GET REGISTER DATA
1412 7650      CMA      /COMPLIMENT TO TEST FOR 7777
1413 5220      SNA CLA      /WAS INPUT REGISTER SET TO 7777
1414 4443      JMP      .+5     /DATA CORRECT, CONTINUE
1415 4011      JMS I      /NO, ERROR
1416 3515      IN3E-1     /"INPUT REGISTER NOT CORRECT"
1417 7777      DH1-1     /"REGISTER DATA"
1420 4063      -1        /NUMBER OF WORDS TO BE OUTPUT
1421 4435      CAF      /INITIALIZE INTERFACE
1422 3021      DBRI      /READ INPUT REGISTER
1423 1021      DCA      DATA1 /SAVE REGISTER DATA
1424 7650      TAD      DATA1
1425 5232      SNA CLA      /DATA CORRECT, CONTINUE
1426 4443      JMP      .+5     /NO, ERROR
1427 3661      JMS I      /"INPUT REGISTER NOT CLEARED"
1430 3515      IN12E-1    /"REGISTER DATA"
1431 7777      DH1-1     /NUMBER OF WORDS TO BE OUTPUT
1432 4444      JMS I      /CHECK FOR LOOP ON CURRENT TEST
1433 5203      JMP      IN1A    /LOOP ON CURRENT TEST

1434 3030      IN2,      /CAN ALL BITS IN INPUT REGISTER BE CLEARED (DBCI)
1435 3031      DCA      /CLEAR ERROR FLAG
1436 4063      CAF      /SET ITERATION COUNT TO 4096(DECIMAL)
1437 7340      CLA CLL CMA /INITIALIZE INTERFACE
1440 0027      AND      /MASK TO TEST ONLY FLIPFLOP BITS
1441 3021      DCA      DATA1 /SAVE MASK
1442 3022      DCA      DATA2 /SAVE EXPECTED RESULT
1443 1021      TAD      DATA1 /GET MASK
1444 4437      DBSO      /BIT SET OUTPUT REGISTER
1445 4434      DBCI      /BIT CLEAR INPUT REGISTER
1446 7300      CLA CLL      /READ INPUT REGISTER
1447 4435      DBRI      /SAVE REGISTER DATA
1450 3023      DCA      DATA3 /COMPARE TO REGISTER DATA
1451 1023      TAD      DATA3 /WERE CORRECT BITS CLEARED
1452 7650      SNA CLA      /DATA CORRECT, CONTINUE
1453 5260      JMP      .+5     /NO, ERROR
1454 4443      JMS I      /"DBCI ERROR"
1455 4003      IN2E-1     /"MASK EXPECTED RECEIVED"
1456 3534      DH3-1

```



```

1457 7775      -3      /NUMBER OF WORDS TO BE OUTPUT
1460 4444      JMS I    XLOOP1  /CHECK FOR LOOP ON CURRENT TEST
1461 5236      JMP      IN2+2  /LOOP ON CURRENT TEST

/ CAN EACH BIT OF INPUT REGISTER BE SET INDEPENDENTLY
/
IN3,          DCA      /CLEAR ERROR FLAG
              TAD      /SET ITERATION COUNT
              DCA      /TO 50(DECIMAL)
              DCA      /CLEAR TEST DATA
              CAF      /INITIALIZE INTERFACE
              TAD      /GET TEST DATA
              DBSO      /BIT SET OUTPUT REGISTER
              CLA CLL

              DBRI      /READ INPUT REGISTER
              DCA      /SAVE REGISTER DATA
              TAD      /GET TEST DATA
              CIA
              TAD      DATA2
              SNA CLA
              JMP      :+5
              JMS I    XERROR
              IN3E-1
              DH4-1
              -2
              JMS I    XLOOP2
              JMP      IN3A
              ISZ      DATA1
              JMP      IN3A
              JMS I    XLOOP1
              JMP      IN3+3
              JMP      IN5

              PAGE
1462 3030
1463 1176
1464 3031
1465 3021
1466 4063
1467 1021
1470 4437
1471 7300
1472 4435
1473 3022
1474 1021
1475 7041
1476 1022
1477 7650
1500 5305
1501 4443
1502 4011
1503 3560
1504 7776
1505 4445
1506 5266
1507 2021
1510 5266
1511 4444
1512 5265
1513 5777
1577 1600
1600

```

```

/VERIFY THAT ALL LATCHING INPUT LINES HOLD DATA
/
IN5,          DCA      /CLEAR ERROR FLAG
              TAD      /SET ITERATION COUNT
              DCA      /TO 50(DECIMAL)
              DCA      /CLEAR TEST DATA
              CAF      /INITIALIZE INTERFACE
              TAD      /GET TEST DATA
              AND      /MASK OFF NON LATCHING BITS
              DCA      /SAVE AS EXPECTED RESULT
              TAD      /GET TEST DATA
              SNA      /ARE ANY BITS TO BE TESTED
              JMP      IN5C
              DBSO
              DBCO
              CLA CLL
              DBRI

              PAGE
1600 3030
1601 1176
1602 3031
1603 3023
1604 4063
1605 1023
1606 0027
1607 3021
1610 1021
1611 7450
1612 5233
1613 4437
1614 4436
1615 7300
1616 4435

```



```

1617 3022 DCA DATA2 /SAVE REGISTER DATA
1620 1021 TAD DATA1 /GET EXPECTED RESULT
1621 7041 CIA
1622 1022 TAD DATA2 /COMPARE TO RECEIVED DATA
1623 7650 SNA CLA /ARE THEY THE SAME
1624 5231 JMP .+5 /DATA CORRECT, CONTINUE
1625 4443 JMS I XERROR /NO, ERROR
1626 4026 IN4E=1 /"LATCH ERROR"
1627 3560 DH4=1 /"EXPECTED RECEIVED"
1630 7776 -2 /NUMBER OF WORDS TO BE OUTPUT
1631 4445 JMS I XLOOP2 /TEST FOR LOOP ON SAME DATA,ESCAPE ON DATA ERROR
1632 5204 JMP IN5A /LOOP WITH SAME DATA
1633 2023 ISZ DATA3 /INCREMENT DATA PATTERN
1634 5204 JMP IN5A /CONTINUE TEST
1635 4444 JMS I XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
1636 5203 JMP IN5+3 /LOOP ON CURRENT TEST
/
IN5C,

```

```

/VERIFY ALL NON LATCHING DATA BITS DO NOT HOLD DATA
/
IN6,
1637 3030 DCA TYPPLG /CLEAR ERROR FLAG
1640 1176 TAD I=62 /SET ITERATION COUNT
1641 3031 DCA LPCNT /TO 50(DECIMAL)
1642 3024 DCA DATA4 /CLEAR TEST DATA
1643 3022 DCA DATA2 /CLEAR EXPECTED RESULT
1644 4003 CAF /INITIALIZE INTERFACE
1645 1027 TAD /GET MASK FOR NON LATCHING BITS
1646 7040 CMA /CHANGE TO MASK OFF LATCHING BITS
1647 0024 AND
1648 0024 AND DATA4
1650 3021 DCA DATA1 /SAVE FOR TRANSMISSION
1651 1021 TAD DATA1 /GET TEST DATA
1652 7450 SNA IN6C /ARE ANY BITS TO BE TESTED
1653 5272 JMP IN6C /NO GET NEXT DATA WORD
1654 4437 DBSO /BIT SET OUTPUT REGISTER
1655 4436 DBCO /BIT CLEAR OUTPUT REGISTER
1656 7300 CLA CLL
1657 4435 DBRI
1660 3023 DCA DATA3 /READ INPUT REGISTER
1661 1023 TAD DATA3 /SAVE REGISTER DATA
1662 7650 SNA CLA /GET RECEIVED DATA
1663 5270 JMP .+5 /DID ANY BITS HOLD DATA
1664 4443 JMS I XERROR /YES, ERROR
1665 4026 -N4E=1 /"LATCH ERROR"
1666 3534 DH3=1 /"MASK EXPECTED RECEIVED"
1667 7775 -3 /NUMBER OF WORDS TO BE OUTPUT
1670 4445 JMS I XLOOP2 /TEST FOR LOOP ON SAME DATA,ESCAPE UN DATA ERROR
1671 5244 JMP IN6A /LOOP WITH SAME DATA
1672 2024 ISZ DATA4 /INCREMENT DATA PATTERN
1673 5244 JMP IN6A /CONTINUE TEST
1674 4444 JMS I XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
1675 5242 JMP IN6+3 /LOOP ON CURRENT TEST
/
IN6C,

```

/VERIFY THAT ALL LATCHING LINES CAN BE CLEARED INDEPENDENTLY



```

1676 3030 IN7, DCA /CLEAR ERROR FLAG
1677 1176 TAD /SET ITERATION COUNT
1700 3031 DCA /TO 50(DECIMAL)
1701 3024 DCA /CLEAR TEST
1702 4063 CAF /INITIALIZE INTERFACE
1703 1027 TAD /GET MASK FOR LATCHING BITS
1704 0024 AND /MASK OFF NON LATCHING BITS
1705 3021 DCA /SAVE FOR TRANSMISSION
1706 3022 DCA /EXPECTED RESULT
1707 1021 TAD /SET OUTPUT REGISTER=7777
1710 4437 DBSO /BIT SET OUTPUT REGISTER
1711 4436 DBCO /BIT CLEAR OUTPUT REGISTER
1712 7300 CLA CLL
1713 1021 TAD DATA1
1714 4434 DBCI
1715 7300 CLA CLL
1716 4435 DBRI
1717 3023 DCA DATA3
1720 1023 TAD DATA3
1721 7650 SNA CLA :+5
1722 5327 JMP XERROR
1723 4443 JMS I
1724 4026 IN4E=1
1725 3534 DH3=1
1726 7775 -3
1727 4445 JMS I
1730 5302 JMP XLOOP2
1731 2024 ISZ IN7A
1732 5302 JMP DATA4
1733 4444 JMS I IN7A
1734 5301 JMP XLOOP1
1735 5777 JMP IN7+3
1777 2000 JMP IN8

```

PAGE

```

2000 3030 IN8, DCA /CLEAR ERROR FLAG
2001 1176 TAD /SET ITERATION COUNT
2002 3031 DCA /TO 50(DECIMAL)
2003 3021 DCA /CLEAR TEST DATA
2004 3022 DCA /CLEAR EXPECTED RESULT
2005 4063 CAF /INITIALIZE INTERFACE
2006 1021 TAD /GET TEST DATA
2007 4434 DBCI /BIT CLEAR INPUT REGISTER
2010 7300 CLA CLL
2011 4435 DBRI
2012 3023 DCA DATA3
2013 1023 TAD DATA3
2014 7650 SNA CLA :+5
2015 5222 JMP XERROR
2016 4443 JMS I

```



```

2017 4003 IN2E=1 /"DBCI ERROR"
2020 3534 DH3=1 /"MASK EXPECTED RECEIVED"
2021 7775 -3 /NUMBER OF DATA WORDS
2022 4445 JMS I XLOOP2 /TEST FOR LOOP WITH SAME DATA
2023 5205 JMP IN8A /LOOP WITH SAME DATA
2024 2021 ISZ DATA1 /INCREMENT DATA PATTERN
2025 5205 JMP IN8A /CONTINUE
2026 4444 JMS I XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
2027 5203 JMP IN8+3 /LOOP ON CURRENT TEST

```

/DOES READING THE INPUT REGISTER  
/CHANGE THE INPUT REGISTER

```

IN9, 3030 DCA TYPFLG /CLEAR ERROR FLAG
2031 1176 TAD I=62 /SET ITERATION COUNT
2032 3031 DCA LPCNT /TO 50 (DECIMAL)
2033 3021 DCA DATA1 /CLEAR TEST DATA
IN9A, 4063 CAF TAD /INITIALIZE INTERFACE
1021 /GET TEST DATA
2036 4437 DBSO /BIT SET OUTPUT REGISTER
2037 7300 CLA CLL
2040 4435 DBRI /READ INPUT REGISTER
2041 7300 CLA CLL
2042 4435 DBRI /READ INPUT REGISTER
2043 3022 DCA DATA2 /SAVE REGISTER DATA
2044 1021 TAD DATA1 /GET TEST DATA
2045 7041 CIA
2046 1022 TAD DATA2
2047 7650 SNA CLA
2050 5255 JMP I XERROR /COMPARE TO REGISTER DATA
2051 4443 JMS I /ARE THEY THE SAME
2052 4034 IN9E=1 /DATA CORRECT, CONTINUE
2053 3560 DH4=1 /NO, ERROR
2054 7776 -2 /"DBRI ERROR"

```

```

2055 4445 JMS I XLOOP2 /"EXPECTED RECEIVED"
2056 5234 JMP IN9A /NUMBER OF DAT WORDS
2057 2021 ISZ DATA1 /TEST FOR LOOP WITH SAME DATA
2060 5234 JMP IN9A /LOOP WITH SAME DATA
2061 4444 JMS I XLOOP1 /INCREMENT DATA PATTERN
2062 5233 JMP IN9+3 /CONTINUE

```

/CHECK FOR LOOP ON CURRENT TEST  
/LOOP ON CURRENT TEST

/DOES CLEARING INPUT REGISTER TWICE SET ANY BIT  
/IN INPUT REGISTER

```

IN10, 3030 DCA TYPFLG /CLEAR ERROR FLAG
2064 1176 TAD I=62 /SET ITERATION COUNT
2065 3031 DCA LPCNT /TO 50 (DECIMAL)
2066 3022 DCA DATA2 /CLEAR EXPECTED RESULT
IN10A, 3021 DCA DATA1 /CLEAR TEST DATA
2070 4063 CAF TAD /INITIALIZE INTERFACE
2071 1021 TAD /GET TEST DATA
2072 4437 DBSO /BIT SET OUTPUT REGISTER
2073 4436 DBCO /BIT CLEAR OUTPUT REGISTER

```



2074	4434	DBCI	/BIT CLEAR INPUT REGISTER
2075	4434	DBCI	/BIT CLEAR INPUT REGISTER
2076	7300	CLA CLL	
2077	4435	DBRI	/READ INPUT REGISTER
2100	3023	DCA	/SAVE REGISTER DATA
2101	1023	TAD	/GET REGISTER DATA
2102	7650	SNA CLA	/IS INPUT REGISTER 0
2103	9310	JMP	/DATA CORRECT, CONTINUE
2104	4443	JMS I	/NO, ERROR
2105	4003	IN2E-1	"DBCI ERROR"
2106	3534	DH3-1	"MASK EXPECTED RECEIVED"
2107	7775	-3	/NUMBER OF DATA WORDS
2110	4445	JMS I	/TEST FOR LOOP WITH SAME DATA
2111	5270	JMP	/LOOP WITH SAME DATA
2112	2021	ISZ	/INCREMENT DATA PATTERN
2113	5270	JMP	/CONTINUE
2114	4444	JMS I	/CHECK FOR LOOP ON CURRENT TEST
2115	5266	JMP	/LOOP ON CURRENT TEST
2116	5777	JMP	/GO TO NEXT TEST
2177	2200		

PAGE

2200	3030	INOU1,	/CLEAR ERROR FLAG
2201	1176	TAD	/SET ITERATION COUNT
2202	3031	DCA	/TO 50 (DECIMAL)
2203	3021	DCA	/CLEAR TEST DATA
2204	3022	DCA	/CLEAR EXPECTED RESULT
2205	4063	CAF	/INITIALIZE INTERFACE
2206	1021	TAD	/GET TEST DATA
2207	4436	DBCO	/BIT CLEAR OUTPUT REGISTER
2210	7300	CLA CLL	
2211	4435	DBRI	/READ INPUT REGISTER
2212	3023	DCA	/SAVE REGISTER DATA
2213	1023	TAD	/GET REGISTER DATA
2214	7650	SNA CLA	/IS OUTPUT REGISTER 0
2215	5222	JMP	/DATA CORRECT, CONTINUE
2216	4443	JMS I	/NO, ERROR
2217	3767	OUT4E-1	"DBCO ERROR"
2220	3534	DH3-1	"MASK EXPECTED RECEIVED"
2221	7775	-3	/NUMBER OF DATA WORDS
2222	4443	JMS I	/TEST FOR LOOP WITH SAME DATA
2223	5205	JMP	/LOOP WITH SAME DATA
2224	2021	ISZ	/INCREMENT DATA PATTERN
2225	5205	JMP	/CONTINUE
2226	4444	JMS I	/CHECK FOR LOOP ON CURRENT TEST
2227	5203	JMP	/LOOP ON CURRENT TEST

  

2200	3030	INOU1,	/WITH BOTH INPUT AND OUTPUT REGISTERS CLEARED
2201	1176	TAD	/DOES CLEARING OUTPUT SET
2202	3031	DCA	/ANY BIT IN INPUT
2203	3021	DCA	
2204	3022	DCA	
2205	4063	CAF	
2206	1021	TAD	
2207	4436	DBCO	
2210	7300	CLA CLL	
2211	4435	DBRI	
2212	3023	DCA	
2213	1023	TAD	
2214	7650	SNA CLA	
2215	5222	JMP	
2216	4443	JMS I	
2217	3767	OUT4E-1	
2220	3534	DH3-1	
2221	7775	-3	
2222	4443	JMS I	
2223	5205	JMP	
2224	2021	ISZ	
2225	5205	JMP	
2226	4444	JMS I	
2227	5203	JMP	

/WITH BOTH INPUT AND OUTPUT REGISTERS CLEARED  
/DOES CLEARING INPUT SET ANY BIT IN OUTPUT



```

2230      3030      INOU2,      DCA      TYPFLG
2231      1176      TAD      C-62
2232      3031      DCA      LPCNT
2233      3021      DCA      DATA1
2234      3022      DCA      DATA2
2235      4063      INOU2A, CAF
2236      1021      TAD      DATA1
2237      4434      DBCI
2240      7300      CLA CLL
2241      4440      DBRO
2242      3023      DCA      DATA3
2243      1023      TAD      DATA3
2244      7650      SNA CLA
2245      5252      JMP      .+5
2246      4443      JMS      XERROR
2247      4003      IN2E-1
2250      3534      DH3-1
2251      7775      -3
2252      4445      JMS      XLOOP2
2253      5235      JMP      INOU2A
2254      2021      ISZ      DATA1
2255      5235      JMP      INOU2A
2256      4444      JMS      XLOOP1
2257      5233      JMP      INOU2+3

```

/WITH THE OUTPUT REGISTER SET TO ALL 1S, AND  
 /THE INPUT REGISTER CLEARED, DOES SELECTIVELY  
 /CLEARING THE OUTPUT REGISTER SET ANY BIT IN  
 /THE INPUT REGISTER

```

/CLEAR ERROR FLAG
/SET ITERATION COUNT
/TO 50 (DECIMAL)
/CLEAR TEST DATA
/CLEAR EXPECTED RESULT
/INITIALIZE INTERFACE
/GET TEST DATA
/8BIT CLEAR INPUT REGISTER

```

```

/READ OUTPUT REGISTER
/SAVE REGISTER DATA
/GET REGISTER DATA
/IS OUTPUT REGISTER 0
/NO, ERROR
/DBCI ERROR
/MASK EXPECTED RECEIVED"
/NUMBER OF DATA WORDS
/TEST FOR LOOP WITH CURRENT DATA
/LOOP WITH SAME DATA
/INCREMENT DATA PATTERN
/CONTINUE
/CHECK FOR LOOP ON CURRENT TEST
/LOOP ON CURRENT TEST

```

```

2260      3030      INOU3,      DCA      TYPFLG
2261      1176      TAD      C-62
2262      3031      DCA      LPCNT
2263      3021      DCA      DATA1
2264      3022      DCA      DATA2
2265      4063      INOU3A, CAF
2266      7040      CMA
2267      4437      DBSO
2270      4434      DBCI
2271      7300      CLA CLL
2272      1027      TAD      FJUMPER
2273      7043      CMA
2274      3022      DCA      DATA2
2275      1021      TAD      DATA1
2276      7040      CMA
2277      0022      AND
2300      3022      DCA      DATA2
2301      1021      TAD      DATA1
2302      4436      DBCO
2303      7300      CLA CLL
2304      4435      DBRI
2305      3023      DCA      DATA3
2306      1023      TAD      DATA3
2307      7041      CIA

```

/WITH THE OUTPUT REGISTER SET TO ALL 1S, AND  
 /THE INPUT REGISTER CLEARED, DOES SELECTIVELY  
 /CLEARING THE OUTPUT REGISTER SET ANY BIT IN  
 /THE INPUT REGISTER

```

/CLEAR ERROR FLAG
/SET ITERATION COUNT
/TO 50 (DECIMAL)
/CLEAR TEST DATA
/CLEAR EXPECTED RESULT
/INITIALIZE INTERFACE
/SET AC=7777
/8BIT SET OUTPUT REGISTER
/8BIT CLEAR INPUT REGISTER
/GET FLIPFLOP JUMPER MASK

```

```

/GET TEST DATA2
/COMPLEMENT
/AND WITH COMPLEMENT OF JUMPER MASK
/TO GET EXPECTED RESULT
/GET TEST DATA
/8BIT CLEAR OUTPUT REGISTER
/READ INPUT REGISTER
/SAVE REGISTER DATA
/GET REGISTER DATA

```



2310	1022	TAD	DATA2	/COMPARE TO EXPECTED RESULT
2311	7650	SNA CLA	.+5	/ARE THEY THE SAME
2312	5317	JMP	XERROR	/DATA CORRECT, CONTINUE
2313	4443	JMS I		/NO, ERROR
2314	3767	OUT4E-1		/DBCO ERROR"
2315	3534	DH3-1		/MASK EXPECTED RECEIVED
2316	7775	-3		/NUMBER OF DATA WORDS
2317	4445	JMS I	XLOOP2	/TEST FOR LOOP WITH SAME DATA
2320	5265	JMP	INOU3A	/LOOP WITH SAME DATA
2321	2021	ISE	DATA1	/INCREMENT DATA PATTERN
2322	5265	JMP	INOU3A	/CONTINUE
2323	4444	JMS I	XLOOP1	/CHECK FOR LOOP ON CURRENT TEST
2324	5263	JMP	INOU3+3	/LOOP ON CURRENT TEST
/				
/WITH THE INPUT REGISTER SET TO ALL 1S, DOES SELECTIVELY				
/CLEARING THE OUTPUT REGISTER CLEAR ANY BITS IN THE INPUT				
/REGISTER (EXCEPT THOSE NOT FLIPFLOPS)				
/				
2325	3030	INOU4,	TYPEFLG	/CLEAR ERROR FLAG
2326	1176	DCA	C=62	/SET ITERATION COUNT
2327	3031	TAD	LPCNT	/TO 50 (DECIMAL)
2330	3021	DCA	DATA1	/CLEAR TEST DATA
2331	4063	INOU4A,	CAF	/INITIALIZE INTERFACE
2332	7040	CMA		/SET AC TO 7777
2333	4437	DBSO		/BIT SET OUTPUT REGISTER
2334	7300	CLA CLL		
2335	1027	TAD	FJUMPER	/GET FLIPFLOP JUMPER MASK
2336	7040	CMA		
2337	0021	AND	DATA1	/COMBINE WITH MASK
2340	7040	CMA		
2341	3022	DCA	DATA2	/TO GET EXPECTED RESULT
2342	1021	TAD	DATA1	/GET TEST DATA
2343	4436	DBCO		/BIT CLEAR OUTPUT REGISTER
2344	7300	CLA CLL		
2345	4435	DBRI		/READ INPUT REGISTER
2346	3023	DCA	DATA3	/SAVE REGISTER DATA
2347	1022	TAD	DATA2	/GET EXPECTED RESULT
2350	7041	CIA		
2351	1023	TAD	DATA3	/COMPARE TO RECEIVED DATA
2352	7650	SNA CLA	.+5	/ARE THEY THE SAME
2353	5360	JMS I	XERROR	/DATA CORRECT, CONTINUE
2354	4443	JMP		/NO, ERROR
2355	3767	OUT4E-1		/DBCO ERROR"
2356	3534	DH3-1		/MASK EXPECTED RECEIVED"
2357	7775	-3		/NUMBER OF DATA WORDS
2360	4445	JMS :	XLOOP2	/TEST FOR LOOP WITH CURRENT DATA
2361	5331	JMP	INOU4A	/LOOP WITH SAME DATA
2362	2021	ISE	DATA1	/INCREMENT DATA PATTERN
2363	5331	JMP	INOU4A	/CONTINUE
2364	4444	JMS I	XLOOP1	/CHECK FOR LOOP ON CURRENT TEST
2365	5330	JMP	INOU4+3	/LOOP ON CURRENT TEST
2366	5777		INT1	/GO TO NEXT TEST
2377	2400			
	2400			



```

/VERIFY THAT EACH BIT SET UP TO SKIP DOES
/
INT1, 3030 DCA TYPFLG /CLEAR ERROR FLAG
      1176 TAD C=62 /SET ITERATION COUNT
      3031 DCA LPCNT /TO 50(DECIMAL)
      3022 DCA DATA2 /INITIALIZE INTERFACE
INT1A, 2404 CAF DATA2
      1022 TAD IJUMPER
      0026 AND
      7450 SNA
      5233 JMP INT1D
      3021 DCA DATA1
      1021 TAD DATA1
      4437 DBSO
      7300 CLA CLL
      4432 DBEI
      4076 SRQ
      5223 JMP INT1AE
      4433 DBSK
      5240 JMP INT1BE
      5231 JMP INT10K
      4433 JMP INT1CE
      5245 JMP XERROR
      4443 JMS I
      4042 INT1E-1
      3515 DH1-1
      7777 -1
INT10K, 2431 JMS I XLOOP2
      5204 JMP INT1A
      2022 ISZ DATA2
      5204 JMP INT1A
      4444 JMS I XLOOP1
      5203 JMP INT1+3
      5252 JMP INT3
      4443 JMP XERROR
      4065 INT3E-1
      3515 DH1-1
      7777 -1
INT1CE, 2443 JMP INT10K
      4443 JMS I XERROR
      4077 INT4E-1
      3515 DH1-1
      7777 -1
      5231 JMP INT10K

/VERIFY THAT EACH BIT NOT JUMPERD TO SKIP DOES NOT
/
INT3, 3030 DCA TYPFLG /CLEAR ERROR FLAG
      1176 TAD C=62 /SET ITERATION COUNT
      3031 DCA LPCNT /TO 50(DECIMAL)
      3022 DCA DATA2 /CLEAR TEST DATA
      4063 CAF /INITIALIZE INTERFACE
INT3A, 2452
      2453
      2454
      2455
      2456

```

2457	1026	TAD	I JUMPER	/GET JUMPER MASK
2460	7040	CMA		/COMPLIMENT FOR NO SKIP BITS
2461	0022	AND	DATA2	/GET BITS TO BE TESTED
2462	7450	SNA		/ARE ANY BITS TO BE TESTED
2463	5300	JMP	INT3C	/NO, GET NEXT DATA PATTERN
2464	3021	DCA	DATA1	/SAVE FOR OUTPUT
2465	1021	TAD	DATA1	/GET TEST DATA
2466	4437	DBSO		/BIT SET OUTPUT REGISTER
2467	7300	CLA CLL		
2470	4433	DBSK		/IS FLAG SET
2471	5276	JMP	.+5	/NO, CONTINUE
2472	4443	JMS I	XERROR	/YES, ERROR
2473	3674	INITSE=1		/SKIP FLAG SET
2474	3515	DH1=1		/REGISTER DATA
2475	7777	-1		
2476	4445	JMS I	XLOOP2	
2477	5256	JMP	INT3A	
2500	2022	ISZ	DATA2	
2501	5256	JMP	INT3A	
2502	4444	JMS I	XLOOP1	
2503	5255	JMP	INT3+3	
2504	5777	JMP	EPASS	

INT3C, /ERROR HANDLER

2577	3257	PAGE		/GET POINTER TO ERROR MESSAGE
2600	0000	ERROR,		/SAVE POINTER
2601	7300	CLA CLL		
2602	1600	TAD I	ERROR	
2603	3234	DCA	MSG	
2604	2200	ISZ	ERROR	
2605	1600	TAD I	ERROR	
2606	3236	DCA	DHDER	
2607	2200	ISZ	ERROR	
2610	1600	TAD I	ERROR	
2611	3264	DCA	DATCNT	
2612	1200	TAD	ERROR	
2613	1377	TAD	(-3	
2614	3776	DCA	LSTDGT	
2615	1776	TAD	LSTDGT	
2616	3266	DCA	ERRAD	
2617	7604	LAS		
2620	0334	AND	SR01	
2621	7540	SZA CLA		
2622	5254	JMP	EHALT	
2623	1030	TAD	TYPFLG	
2624	7640	SZA CLA		
2625	5241	JMP	DATOUT	
2626	7040	CMA		
2627	3030	DCA	TYPFLG	
2630	4775	JMS	OCTASC	
2631	4441	JMS I	XPRINT	
2632	3024	ERADR=1		

/CLEAR ERROR FLAG



/NUMBER OF WORDS TO BE OUTPUT

```

2633 4441 JMS I XPRINT
2634 0000 MSG,
2635 4441 JMS I XPRINT
2636 0000 DHDER,
2637 4441 JMS I XPRINT
2640 3512 CRLF-1
2641 1264 DATOUT, TAD DATCNT
2642 7650 SNA CLA
2643 5254 JMP EHALT
2644 1265 DATAP
2645 3010 DCA POINT1
2646 1410 TAD I POINT1
2647 4774 JMS BITOUT
2650 2264 ISZ DATCNT
2651 5246 JMP BITS
2652 4441 JMS I XPRINT
2653 3512 CRLF-1
2654 7604 LAS
2655 0333 AND SR00
2656 7640 SZA CLA
2657 5262 JMP .+3
2660 1266 TAD ERRAD
2661 7402 HLT
2662 2200 ISZ ERROR
2663 5600 JMP I ERROR
2664 0000 DATCNT,
2665 0020 DATAP,
2666 0000 ERRAD,

```

/TEST FOR LOOP ON CURRENT TEST

```

2667 0000 LOOP1,
2670 1030 TAD TYPFLG
2671 7650 SNA CLA
2672 5277 JMP LP1EXA
2673 7604 LAS
2674 0341 AND SR06
2675 7640 SZA CLA
2676 5310 JMP LP1EXX-1
2677 7604 LAS
2680 0340 AND SR05
2681 7610 SZA CLA
2682 5305 JMP LP1EXT
2683 2031 ISZ LPCNT
2684 5310 JMP LP1EXX
2685 7604 LAS
2686 0335 AND SR02
2687 7650 SNA CLA
2690 2267 ISZ LOOP1
2691 5667 JMP I LOOP1
2692 0000 LP1EXX,
2693 1030 TAD TYPFLG

```

/TEST FOR LOOP ON CURRENT DATA

```

2712 0000 LOOP2,
2713 1030 TAD TYPFLG

```

2714 7650 SNA CLA LP2EXT  
2715 5326 JMP  
2716 7604 LAS  
2717 0341 AND SR06  
2720 7650 SNA CLA  
2721 5326 .+5  
2722 1312 LOOP2  
2723 1373 (5  
2724 3312 LOOP2  
2725 5712 LOOP2  
2726 7604 JMP I  
2727 0336 AND SR03  
2730 7650 SNA CLA  
2731 2312 ISZ LOOP2  
2732 5712 JMP I LOOP2  
2733 4000  
2734 2000  
2735 1000  
2736 0400  
2737 0200  
2740 0100  
2741 0040

/DATA CORRECT, CONTINUE

/OCTAL TO PACKED ASCII CONVERSION

2773 0005  
2774 3031  
2775 3000  
2776 3027  
2777 7775  
3000 0000  
3001 7300  
3002 1227  
3003 4046  
3004 4212  
3005 3226  
3006 1227  
3007 4212  
3010 3227  
3011 5600  
3012 0000  
3013 6377  
3014 3114  
3015 4119  
3016 7106  
3017 7004  
3020 0376  
3021 4115  
3022 0376  
3023 1375  
3024 5612  
3025 3736  
3026 4040

PAGE /  
OCTASC. 0  
CLA CLL LSTDGT  
TAD BSW  
JMS  
DCA  
TAD  
JMS  
DCA  
JMP I  
0  
AND  
MQL  
MGA  
CLL RTL  
RAL  
AND  
MGA  
AND  
TAD  
JMP I  
TEXT /

/GET WORD TO BE CONVERTED  
/SWAP HALVES, SEPARATE DIGITS,  
/CONVERT MOST SIGNIFICANT  
/DIGITS TO ASCII  
/CONVERT LEAST SIGNIFICANT  
/DIGITS TO ASCII  
/RETURN



3027 4040  
3030 4000

```

/OUTPUT 12 BIT BINARY WORD
/
BITOUT, 0
MQL
TAD
DCA
MCA RAL
MQL
TAD
SNL
AND
JMS I
ISZ
JMP
TAD
JMS I
TAD
JMS I
JMP I BITOUT

/SAVE DATA IN MQ
/SET UP TO OURPUT
/12 BITS
/GET DATA
/GET MSB INTO LINK
/SAVE REST OF WORD
/GET ASCII 1 INTO AC

/IS BIT=1
/NO, CHANGE TO ASCII 0
/OUTPUT BIT
/CONTINUE
/TYPE 2 SPACES
/AFTER LAST BIT HAS BEEN
/OUTPUTTED
/RETURN
    
```

/CHARACTER STRING OUTPUT ROUTINE

3171 0240  
3172 0260  
3173 0261  
3174 7764  
3175 6060  
3176 0707  
3177 0077

```

PAGE, 0
PRINT, 0
CLA CLL
TAD I
DCA
ISZ
TAD I
MQL
MCA
3SW
JMS
MCA
JMS
JMP

/GET POINTER TO MESSAGE
/SET UP AUTO-INDEX REGISTER
/SET UP RETURN
/GET PACKED WORD
/SAVE IN MQ
/GET WORD
/SHAP HALVES
/DECODE AND OUTPUT
/GET WORD
/DECODE AND OUTPUT
/CONTINUE

TYPSET, 0
AND
SNA
JMP I PRINT

/UNPACK, DECODE, OUTPUT
/
TYPSET, 0
AND
SNA
JMP I PRINT
    
```

3215 0000  
3216 0243  
3217 7450  
3220 5600

/MASK UNWANTED BITS  
/IS AC=0  
/YES, END OF MESSAGE, EXIT

3221	1244	TAD	M40	/SUBTRACT 40
3222	7510	SPA		/IS PACKED CHARACTER >40
3223	5226	JMP	.+3	/NO
3224	1250	TAD	K240	/YES, CONVERT TO ASCII
3225	5241	JMP	MTP	/OUTPUT
3226	7001	IAC		/ADD 1 TO AC
3227	7440	SZA		/IS CHARACTER=37
3230	5233	JMP	.+3	/NO
3231	1245	TAD	K215	/GET CODE FOR CARRIAGE RETURN
3232	5241	JMP	MTP	/OUTPUT
3233	7001	IAC		/ADD 1 TO AC
3234	7440	SZA		/IS CHARACTER=37
3235	5240	JMP	.+3	/NO
3236	1246	TAD	K212	/GET CODE FOR LINE FEED
3237	5241	JMP	MTP	/OUTPUT
3240	1247	TAD	K336	/PACKED CHARACTER >40, CONVERT TO ASCII
3241	4442	JMS I	XTYPE	/OUTPUT
3242	5615	JMP I	TYPSET	
3243	0077	77		
3244	7740	=40		
3245	0215	215		
3246	0212	212		
3247	0336	336		
3250	0240	240		
		/		/OUTPUT ONE CHARACTER TO TTY
		/		
3251	0000	0		
3252	6046	TLS		
3253	6041	TSF	.-1	
3254	5253	JMP		
3255	7200	CLA		
3256	5651	JMP I	TYPE	
		/		
		/		
		/		
3257	7604	LAS		
3260	0777	AND	SR04	
3261	7640	SZA	CLA	
3262	5776	JMP	INIT1	
3263	7604	LAS		
3264	0775	AND	SR05	
3265	7640	SZA	CLA	
3266	5272	JMP	EPAS1	
3267	4441	JMS I	XPRINT	
3270	3275	HEP-1		
3271	5776	JMP	INIT1	
3272	1374	TAD	(207	
3273	4251	JMS	TYPE	
3274	5776	JMP	INIT1	
3275	3736	HEP,	/-DR/	
3276	0422			
3277	0000			



/TELETYPE MESSAGES

3374 0207  
3375 2740  
3376 0400  
3377 2737

PAGE  
M1,

TEXT /-SET SR FOR DEVICE CODE AND CONT/

3400 3736  
3401 2305  
3422 2440  
3403 2322  
3404 4006  
3405 1722  
3406 4004  
3407 0526  
3410 1103  
3411 0540  
3412 0317  
3413 0405  
3414 4001  
3415 1604  
3416 4003  
3417 1716  
3420 2400

M2,

TEXT /-SET SR FOR INTERRUPT JUMPERS AND CONT/

3421 3736  
3422 2305  
3423 2440  
3424 2322  
3425 4006  
3426 1722  
3427 4011  
3430 1624  
3431 0522  
3432 2225  
3433 2024  
3434 4012  
3435 2515  
3436 2005  
3437 2223  
3440 4001  
3441 1604  
3442 4003  
3443 1716  
3444 2400

M2A,

TEXT /-SET SWITCHES FOR FLIPFLOP JUMPERS AND CONTINUE/

3445 3736  
3446 2305  
3447 2440  
3450 2327  
3451 1124  
3452 0310  
3453 0523  
3454 4006  
3455 1722  
3456 4006

3457 1411  
3460 2006  
3461 1417  
3462 2040  
3463 1225  
3464 1520  
3465 0522  
3466 2340  
3467 0116  
3470 0440  
3471 0317  
3472 1624  
3473 1116  
3474 2505  
3475 0000  
3476 3736  
3477 2305  
3500 2440  
3501 2322  
3502 4006  
3503 1722  
3504 4022  
3505 2516  
3506 4001  
3507 1604  
3510 4003  
3511 1716  
3512 2400  
3513 3736  
3514 0000

M3, TEXT /--SET SR FOR RUN AND CONT/

CRLF, TEXT /--/

/DATA HEADERS  
/

DH0, /--REGISTER DATA/  
DH1, /

DH2, TEXT /--AC CONTENTS/

DH3, TEXT /--MASK EXPECTED RECEIVED/

3515 0000  
3516 3736  
3517 2205  
3520 0711  
3521 2324  
3522 0522  
3523 4004  
3524 0124  
3525 0100  
3526 3736  
3527 0103  
3530 4003  
3531 1716  
3532 2405  
3533 1624  
3534 2300  
3535 3736  
3536 1501  
3537 2313  
3540 4040  
3541 4040  
3542 4040



3543 4040  
 3544 4040  
 3545 0530  
 3546 2005  
 3547 0324  
 3550 0504  
 3551 4040  
 3552 4040  
 3553 4040  
 3554 2205  
 3555 0305  
 3556 1126  
 3557 0504  
 3560 0000  
 3561 3736  
 3562 0530  
 3563 2005  
 3564 0324  
 3565 0504  
 3566 4040  
 3567 4040  
 3570 4040  
 3571 2205  
 3572 0305  
 3573 1126  
 3574 0504  
 3575 0000  
 3576 3736  
 3577 2205  
 3600 0711  
 3601 2324  
 3602 0522  
 3603 4040  
 3604 4040  
 3605 4040  
 3606 4004  
 3607 0124  
 3610 0140  
 3611 1725  
 3612 2440  
 3613 4040  
 3614 4040  
 3615 4040  
 3616 0401  
 3617 2401  
 3620 4011  
 3621 1600  
 3622 3736  
 3623 0103  
 3624 4003  
 3625 1716  
 3626 2405  
 3627 1624  
 3630 2340  
 3631 4040

DH4, TEXT /-EXPECTED RECEIVED/

DH5, TEXT /-REGISTER DATA OUT DATA IN/

DH6, TEXT /-AC CONTENTS DATA OUT DATA IN/

3632 4004  
3633 0124  
3634 0140  
3635 1725  
3636 2440  
3637 4040  
3640 4040  
3641 4040  
3642 0401  
3643 2401  
3644 4011  
3645 1600

/ERROR MESSAGE  
/

INIT1E, TEXT /OUTPUT REG NOT CLEARED/

3646 1725  
3647 2420  
3650 2524  
3651 4022  
3652 0507  
3653 4016  
3654 1724  
3655 4003  
3656 1405  
3657 0122  
3660 0504  
3661 0000  
3662 1116  
3663 2025  
3664 2440  
3665 2205  
3666 0740  
3667 1617  
3670 2440  
3671 0314  
3672 0501  
3673 2205  
3674 0400  
3675 2313  
3676 1120  
3677 4006  
3700 1401  
3701 0740  
3702 2305  
3703 2400  
3704 2402  
3705 2217  
3706 4004  
3707 1104  
3710 4016  
3711 1724  
3712 4003  
3713 1405  
3714 0122  
3715 4001

INIT2E, TEXT /INPUT REG NOT CLEARED/

INIT3E, TEXT /SKIP FLAG SET/

TRAN1E, TEXT /DSDR DID NOT CLEAR AC/



3716	0300		
3717	0402	TRAN2E, TEXT	/DBRI DID NOT CLEAR AC/
3720	2211		
3721	4004		
3722	1104		
3723	4016		
3724	1724		
3725	4003		
3726	1405		
3727	0122		
3730	4001		
3731	0300		
3732	0402	TRAN3E, TEXT	/DBSO CHANGED AC/
3733	2317		
3734	4003		
3735	1001		
3736	1607		
3737	0504		
3740	4001		
3741	0300		
3742	0402	TRAN4E, TEXT	/DBCO CHANGED AC/
3743	0317		
3744	4003		
3745	1001		
3746	1607		
3747	0504		
3750	4001		
3751	0300		
3752	0402	TRAN5E, TEXT	/DBCI CHANGED AC/
3753	0311		
3754	4003		
3755	1001		
3756	1607		
3757	0504		
3760	4001		
3761	0300		
3762	0402	OUT1E, TEXT	/DBSO ERROR/
3763	2317		
3764	4005		
3765	2222		
3766	1722		
3767	0000		
3770	0402	OUT4E, TEXT	/DBCO ERROR/
3771	0317		
3772	4005		
3773	2222		
3774	1722		
3775	0000		
3776	0402	OUT7E, TEXT	/DBRO ERROR/
3777	2217		
4000	4005		
4001	2222		
4002	1722		
4003	0000		
4004	0402	IN2E, TEXT	/DBCI ERROR/

4005	0311		
4006	4005		
4007	2222		
4010	1722		
4011	0000		
4012	1116		
4013	2025		
4014	2440		
4015	2205		
4016	0711		
4017	2324		
4020	0522		
4021	4004		
4022	0124		
4023	0140		
4024	0522		
4025	2217		
4026	2200		
4027	1401		
4030	2403		
4031	1040		
4032	0522		
4033	2217		
4034	2200		
4035	0402		
4036	2211		
4037	4005		
4040	2222		
4041	1722		
4042	0000		
4043	1116		
4044	2405		
4045	2222		
4046	2520		
4047	2440		
4050	0103		
4051	2411		
4052	2605		
4053	0000		
4054	1617		
4055	4011		
4056	1624		
4057	2222		
4060	2225		
4061	2024		
4062	5440		
4063	2313		
4064	1120		
4065	0000		
4066	1116		
4067	2405		
4070	2222		
4071	2520		
4072	2454		
4073	4016		

  

IN3E,	TEXT	/INPUT REGISTER DATA ERROR/
IN4E,	TEXT	/LATCH ERROR/
IN9E,	TEXT	/DBRI ERROR/
INT1E,	TEXT	/INTERRUPT ACTIVE/
INT2E,	TEXT	/NO INTERRUPT, SKIP/
INT3E,	TEXT	/INTERRUPT, NO SKIP/



4074	1740		
4075	2313		
4076	1120		
4077	0000		
4100	1617	INT4E, TEXT	/NO INTERRUPT, NO SKIP/
4101	4011		
4102	1624		
4103	0522		
4104	2225		
4105	2024		
4106	5440		
4107	1617		
4110	4023		
4111	1311		
4112	2000	INT5E, TEXT	/NO SKIP/
4113	1617		
4114	4023		
4115	1311		
4116	2000		
4117	2313	INT6E, TEXT	/SKIP/
4120	1120		
4121	0000		

S

0176	7716
0177	0106





400 1111111 1111111 1111111 1111111 1111111 1111111 1111111 1111111  
410 1111111 1111111 1111111 1111111 1111111 1111111 1111111 1111111

420  
430

440  
450

460  
470

500  
510

520  
530

540  
550

560  
570

600  
610

620  
630

640  
650

660  
670

700  
710

720  
730

740  
750

760  
770

BIT01	3035	IN2	1434	K212	3246	SR00	2733
BITOUT	3031	IN2E	4004	K215	3245	SR01	2734
BITS	2646	IN3	1462	K240	3250	SR02	2735
BSW	4046	IN3A	1466	K336	3247	SR03	2736
CAF	4063	IN3E	4012	K7700	0062	SR04	2737
CNTR1	0020	IN4E	4027	LOOP1	2667	SR05	2740
CRLF	3513	IN5	1600	LOOP2	2712	SR06	2741
DATA1	0021	IN5A	1604	LP1EXA	2677	SR0	4076
DATA2	0022	IN5C	1633	LP1EXT	2705	START1	0202
DATA3	0023	IN6	1637	LP1EXX	2711	START2	0244
DATA4	0024	IN6A	1644	LP2EXT	2726	T1OT	0250
DATA5	0025	IN6C	1672	LPCNT	0031	TRAN1	0452
DATAP	2665	IN7	1676	LSTDGT	3027	TRAN1E	3704
DATCNT	2664	IN7A	1702	M1	3400	TRAN2	0471
DATOUT	2641	IN7C	1731	M2	3421	TRAN2E	3717
DBCI	4434	IN8	2000	M2A	3445	TRAN3	0510
DBCI X	0300	IN8A	2005	M3	3476	TRAN3E	3732
DBCO	4436	IN9	2030	M40	3244	TRAN4	0530
DBCOX	0312	IN9A	2034	MEP	3275	TRAN4E	3742
DBDIX	0261	IN9E	4035	MESG	2634	TRAN5	0550
DBEI	4432	INIT1	0400	M0	0114	TRAN5E	3752
DBEIX	0266	INIT1E	3646	MQA	4115	TRAN6	0600
DBRI	4435	INIT2	0416	MQL	3114	TRAN7	0616
DBRIX	0305	INIT2E	3662	MSTDGT	3026	TRAN8	0634
DBRO	4440	INIT3	0435	MTP	3241	TYPE	0113
DBROX	0324	INIT3E	0675	OC1ASC	3000	TYPFLG	3251
DBSK	4433	INOUI	2200	OUT1	1000	TYPSET	0030
DBSKX	0273	INOUIA	2205	OUT1E	3762	XOBCI	3215
DBSO	4437	INOUI2	2230	OUT2	1033	XOBCO	0034
DBSOX	0317	INOUI2A	2235	OUT3	1054	XOBEI	0036
DH0	3515	INOUI3	2260	OUT3A	1060	XOERI	0032
DH1	3516	INOUI3A	2265	OUT4	1105	XOERO	0040
DH2	3526	INOUI4	2325	OUT4A	1111	XOBSK	0033
DH3	3535	INOUI4A	2331	OUT4E	3770	XOBSO	0037
DH4	3561	INT1	2400	OUT5	1200	XERROR	0043
DH5	3576	INT1A	2404	OUT5A	1205	XLOOP1	0044
DH6	3622	INT1AE	2423	OUT6	1230	XLOOP2	0045
DHDER	2636	INT1BE	2440	OUT6A	1234	XPRINT	0041
D1OT	0260	INTICE	2445	OUT7	1262	XTYPE	0042
EHALT	2634	INTID	2433	OUT7A	1266		
EPAS1	3272	INT1E	4043	OUT7E	3776		
EPASS	3257	INT1OK	2431	OUT8	1315		
ERRADR	3025	INT1E	4054	OUT8A	1322		
ERRAD	2336	INT3	2452	PRINT1	0233		
ERROR	2600	INT3A	2456	POINT1	0010		
ERROR	0527	INT3C	2500	PRINT	0000		
IJUMPE	0026	INT3E	4066	RETURN	0106		
IN1	1400	INT4E	4100	SIMBSH	0046		
IN10	2063	INT5E	4113	SIMCAF	0063		
IN10A	2070	INT6E	4117	SIMMOA	0115		
IN1A	1403	IOTS	0247	SIMSRQ	0076		
IN1B	1420	K0077	3243	SPLIT	3012		



ERRORS DETECTED: 0  
LINKS GENERATED: 10  
RUN-TIME: 10 SECONDS  
3K CORE USED

